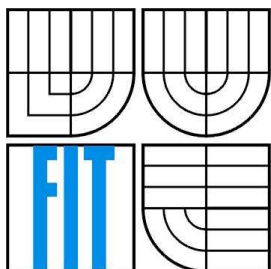


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

AUTOMATOVÉ SYSTÉMY

AUTOMATA SYSTEMS

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

ANDREJ KAŠČÁK

VEDOUCÍ PRÁCE
SUPERVISOR

prof. RNDr. ALEXANDER MEDUNA, CSc

BRNO 2008

Vysoké učení technické v Brně - Fakulta informačních technologií

Ústav informačních systémů

Akademický rok 2007/2008

Zadání bakalářské práce

Řešitel: **Kaščák Andrej**
Obor: Informační technologie
Téma: **Automatové systémy**
Kategorie: Teorie informatiky
Pokyny:

1. Seznamte se detailně s automatovými systémy.
2. Zaveďte vlastní typ automatových systémů.
3. Studujte vlastnosti zavedených systémů. Porovnejte je s vlastnostmi jiných automatových systémů.
4. Studujte vhodné aplikace navržených systémů (např. kompilátory, lingvistika, mikrobiologie).
5. Zhodnoťte dosažené výsledky a diskutujte další možný vývoj projektu.

Literatura:

- Meduna, A.: Automata and Languages, Springer, London, 2000

Při obhajobě semestrální části projektu je požadováno:

- Body 1-2.

Podrobné závazné pokyny pro vypracování bakalářské práce naleznete na adrese
<http://www.fit.vutbr.cz/info/szz/>

Technická zpráva bakalářské práce musí obsahovat formulaci cíle, charakteristiku současného stavu, teoretická a odborná východiska řešených problémů a specifikaci etap (20 až 30% celkového rozsahu technické zprávy).

Student odevzdá v jednom výtisku technickou zprávu a v elektronické podobě zdrojový text technické zprávy, úplnou programovou dokumentaci a zdrojové texty programů. Informace v elektronické podobě budou uloženy na standardním nepřepisovatelném paměťovém médiu (CD-R, DVD-R, apod.), které bude vloženo do písemné zprávy tak, aby nemohlo dojít k jeho ztrátě při běžné manipulaci.

Vedoucí: **Meduna Alexander, prof. RNDr., CSc., UIFS FIT VUT**

Datum zadání: 1. listopadu 2007

Datum odevzdání: 14. května 2008

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
Fakulta informačních technologií
Ústav informačních systémů
602 00 Brno, Božetěchova 2

doc. Ing. Jaroslav Zendulka, CSc.
vedoucí ústavu

**LICENČNÍ SMLOUVA
POSKYTOVANÁ K VÝKONU PRÁVA UŽÍT ŠKOLNÍ DÍLO**

uzavřená mezi smluvními stranami

1. Pan

Jméno a příjmení: **Andrej Kaščák**
Id studenta: 78944
Bytem: Gaštanová 39, 010 01 Žilina
Narozen: 03. 10. 1986, Žilina
(dále jen "autor")

a

2. Vysoké učení technické v Brně

Fakulta informačních technologií
se sídlem Božetěchova 2/1, 612 66 Brno, IČO 00216305
jejímž jménem jedná na základě písemného pověření děkanem fakulty:

.....
(dále jen "nabyvatel")

**Článek 1
Specifikace školního díla**

1. Předmětem této smlouvy je vysokoškolská kvalifikační práce (VŠKP):
bakalářská práce

Název VŠKP: Automatové systémy
Vedoucí/školicel VŠKP: Meduna Alexander, prof. RNDr., CSc.
Ústav: Ústav informačních systémů
Datum obhajoby VŠKP:

VŠKP odevzdal autor nabyvateli v:

tištěné formě	počet exemplářů: 1
elektronické formě	počet exemplářů: 2 (1 ve skladu dokumentů, 1 na CD)

2. Autor prohlašuje, že vytvořil samostatnou vlastní tvůrčí činností dílo shora popsané a specifikované. Autor dále prohlašuje, že při zpracovávání díla se sám nedostal do rozporu s autorským zákonem a předpisy souvisejícími a že je dílo dílem původním.
3. Dílo je chráněno jako dílo dle autorského zákona v platném znění.
4. Autor potvrzuje, že listinná a elektronická verze díla je identická.

Článek 2

Udělení licenčního oprávnění

1. Autor touto smlouvou poskytuje nabyvateli oprávnění (licenci) k výkonu práva uvedené dílo nevýdělečně užit, archivovat a zpřístupnit ke studijním, výukovým a výzkumným účelům včetně pořizování výpisů, opisů a rozmnožení.
2. Licence je poskytována celosvětově, pro celou dobu trvání autorských a majetkových práv k dílu.
3. Autor souhlasí se zveřejněním díla v databázi přístupné v mezinárodní síti:
 - ☐ ihned po uzavření této smlouvy
 - ☐ 1 rok po uzavření této smlouvy
 - ☐ 3 roky po uzavření této smlouvy
 - ☐ 5 let po uzavření této smlouvy
 - ☐ 10 let po uzavření této smlouvy(z důvodu utajení v něm obsažených informací)
4. Nevýdělečné zveřejňování díla nabyvatelem v souladu s ustanovením § 47b zákona č. 111/1998 Sb., v platném znění, nevyžaduje licenci a nabyvatel je k němu povinen a oprávněn ze zákona.

Článek 3


Závěrečná ustanovení

1. Smlouva je sepsána ve třech vyhotoveních s platností originálu, přičemž po jednom vyhotovení obdrží autor a nabyvatel, další vyhotovení je vloženo do VŠKP.
2. Vztahy mezi smluvními stranami vzniklé a neupravené touto smlouvou se řídí autorským zákonem, občanským zákoníkem, vysokoškolským zákonem, zákonem o archivnictví, v platném znění a popř. dalšími právními předpisy.
3. Licenční smlouva byla uzavřena na základě svobodné a pravé vůle smluvních stran, s plným porozuměním jejímu textu i důsledkům, nikoliv v tísní a za nápadně nevýhodných podmínek.
4. Licenční smlouva nabývá platnosti a účinnosti dnem jejího podpisu oběma smluvními stranami.

V Brně dne:

.....

Nabyvatel


.....

Autor

Abstrakt

Táto bakalárska práca sa zaoberá automatovými systémami, konkrétne definuje stavovo a pravidlovo kontrolované paralelné typy automatových systémov, ktorých komponentmi sú konečné automaty, na základe rozboru a skúmania princípov už existujúcich systémov. Taktiež sú práci porovnávané tieto nové systémy s tými existujúcimi, sú skúmané ich vlastnosti a možnosť transformácie stavovo kontrolovaného paralelného automatového systému na pravidlovo kontrolovaný paralelný automatový systém a aj opačne.

Kľúčové slová

Automatový systém, paralelné automatové systémy, paralelné komunikujúce automatové systémy, stavovo kontrolované paralelné automatové systémy, pravidlovo kontrolované paralelné automatové systémy.

Abstract

This bachelor thesis deals with automata systems, especially the ones, which are defined as state and or rule controlled parallel automata systems. Their components are finite automata systems. Definitions are retrieved from analyzing and researching already existing automata systems. In the thesis, a comparison of these systems is made with systems defined a priori. The thesis also analyzes first features of these systems, and second a possibilities of transforming the state controlled parallel automata system into the rule controlled parallel automata system and vice versa.

Keywords

Automata system, parallel automata systems, parallel communicating automata systems, state controled parallel automata systems, rule controled parallel automata systems.

Citácie

Kaščák Andrej: Automatové systémy. Brno, 2008, bakalárska práca, FIT VUT v Brne.

Automatové systémy

Prehlásenie

Prehlasujem, že som túto bakalárskou prácu vypracoval samostatne pod vedením
prof. RNDr. Alexandra Meduny, CSc.

Uviedol som všetky literárne pramene a publikácie, z ktorých som čerpal.

.....
Andrej Kaščák
12.5.2008

Pod'akovanie

Rád by som na tomto mieste poďakoval môjmu vedúcemu prof. RNDr. Alexandrovi Medunovi, CSc
za odbornú pomoc, ochotu a čas, ktorý mi pri tvorbe práce venoval.

© Andrej Kaščák, 2008.

Táto práca vznikla ako školské dielo na Vysokom učení technickom v Brne, Fakulte informačných technológií. Práca je chránená autorským zákonom a jej použitie bez udelenia oprávnenia autorom je nezákonné, s výnimkou zákonom definovaných prípadov.

Obsah

Obsah	1
Úvod	2
1 Základné pojmy a definície	3
1.1 Základná terminológia	3
1.2 Automatové systémy	4
1.3 Kanonické multigeneratívne gramatické systémy	7
1.3.1 Kanonický n-generatívny nonterminálovo synchronizovaný gramatický systém	8
1.3.2 Kanonický n-generatívny pravidlovo synchronizovaný gramatický systém	8
2 Definovanie nových automatových systémov	9
2.1 Stavovo kontrolovaný paralelný automatový systém	9
2.2 Pravidlovo kontrolovaný paralelný automatový systém	11
2.3 Vzťah medzi n-SKP a n-PKP	13
2.3.1 Prevod n-SKP na ekvivalentný n-PKP	14
2.3.2 Prevod n-PKP na ekvivalentný n-SKP	16
2.4 Vlastnosti n-SKP a n-PKP a porovnanie s ostatnými automatovými systémami	18
3 Budúci vývoj a uplatnenie v praxi	21
Záver	22
Literatúra	23
Zoznam príloh	24

Úvod

Táto práca sa svojím obsahom snaží rozšíriť oblasť teórie formálnych jazykov, ktorá je dôležitou časťou informatiky a v súčasnosti zahŕňa poznatky, ktoré sa v praxi využívajú napríklad na určenie možností a obmedzení algoritmicky riešených problémov (napríklad odhaľujú nerozhodnuteľnosť algoritmov a podobne) a teda najmä v oblasti programovacích jazykov, ale aj v oblastiach prekladačov, umelej inteligencie, riadiacej techniky a počítačovej grafiky a dokonca aj v netechnických odboroch, ako sú napríklad biológia a genetika. Táto teória pracuje s dvoma základnými nástrojmi a to s gramatikou, ktorá umožňuje popísať štruktúru viet formálneho jazyka a s automatmi, ktoré sú schopné túto štruktúru identifikovať.

Táto bakalárska práca si kladie za cieľ preskúmať a preštudovať niektoré už existujúce automatové systémy, v krátkosti ukázať ako tieto systémy fungujú a na základe pochopenia princípov navrhnuť nový typ automatového systému.

V bakalárskej práci sú definované dva nové typy automatových systémov, ktoré sú založené na paralelnom princípe, ktorý je v súčasnosti veľmi často používaný a to najmä preto, lebo dokáže pomocou viacerých jednoduchých štruktúr dosiahnuť taký istý výsledok, ako pri použití jednej zložitej štruktúry.

Tieto systémy pracujú tak, že vstupný reťazec je rozdelený na viac častí a jednotlivé časti sú paralelne spracovávané jednotlivými komponentmi, pričom sa kontrolujú buď stavy, do ktorých sa automaty dostanú, alebo pravidlá, ktoré sú použité pri prechodoch.

Prvá kapitola je všeobecný úvod, v ktorom je naznačená ústredná myšlienka práce a rozdelenie práce na jednotlivé časti.

V druhej kapitole sú pripomenuté základné pojmy a definície, ktoré je dôležité poznať pre pochopenie ďalších častí práce a tieto pojmy sú neskôr použité či už pri definíciách, alebo dôkazoch v nasledujúcich kapitolách. Sú v nej tiež ukázané základné princípy a metódy niektorých už definovaných automatových systémov, grafické ukážky týchto automatových systémov, pričom tieto systémy budú v závere porovnané s novo definovanými systémami. Taktiež sú v tejto kapitole ukázané základné vlastnosti kanonických multigeneratívnych gramatických systémov, ktorých princípy činnosti sú využité pri tvorbe nových automatových systémov v tretej kapitole.

Tretia kapitola je venovaná definovaniu nových typov automatových systémov, konkrétne sú v nej definované najskôr stavovo kontrolované paralelné automatové systémy a následne pravidlovo kontrolované paralelné automatové systémy. Tieto systémy sú v tejto kapitole študované, sú zobrazené vzorové príklady ich činnosti, následne sú ukázané algoritmy, pomocou ktorých je možné stavovo kontrolovaný systém transformovať na pravidlovo kontrolovaný systém a aj naopak a v závere kapitoly sú porovnávané so systémami z druhej kapitoly.

Štvrtá kapitola sa venuje možnému budúcemu výskumu a rozvoju na poli automatových systémov, ktoré boli definované v tretej kapitole a ich možnému praktickému využitiu.

Posledná piata kapitola, záver, je venovaná zhrnutiu celého obsahu práce, dosiahnutých výsledkov a uhlu pohľadu autora na to, čo mu táto práca priniesla.

1 Základné pojmy a definície

V tejto kapitole je uvedená základná terminológia, ktorá je ďalej v práci používaná. Sú tu uvedené iba úplne najzákladnejšie definície a pre podrobnejšie informácie je vhodné si pozrieť odkazy na príslušnú literatúru [1], [2].

Sú tu taktiež uvedené príklady už definovaných automatových systémov, s ktorým budú novo definované systémy neskôr porovnávané a tiež základné definície a princípy kanonických multigeneratívnych systémov.

1.1 Základná terminológia

Najskôr upresním pojem abeceda, abeceda je neprázdna množina prvkov, ktoré nazývame symboly a táto množina je konečná. Označuje sa symbolom Σ a potom symbolika Σ^* značí množinu všetkých reťazcov nad touto abecedou Σ . Reťazec, ktorý neobsahuje žiadny symbol a teda môžeme povedať prázdny reťazec sa označuje ε a symbolika Σ^+ značí $\Sigma^* - \{\varepsilon\}$.

Platí, že ak x je reťazec nad abecedou Σ a ε je reťazec nad abecedou Σ tak potom aj reťazec $x\varepsilon$ je reťazec nad abecedou Σ .

Teraz nasleduje definícia konečného automatu.

Definícia 1.1.1

Konečný automat je päťica symbolov:

$$M = (Q, \Sigma, R, s, F),$$

kde Q je konečná množina stavov v ktorých sa môže automat nachádzať, Σ je vstupná abeceda prvkov, ktorá je prijímaná automatom, R je konečná množina pravidiel tvaru $pa \rightarrow q$, kde $p, q \in Q$, $a \in \Sigma \cup \{\varepsilon\}$, s je počiatočný stav, platí $s \in Q$ a F je konečná množina koncových stavov, pričom platí $F \subseteq Q$.

Definícia 1.1.2

Konfigurácia konečného automatu M je reťazec $\chi \in Q\Sigma^*$, teda je to informácia o tom v ktorom stave sa aktuálne automat nachádza a o reťazci, ktorý je na vstupe tohoto automatu.

Definícia 1.1.3

Prechod konečného automatu je prečítanie jedného prvku zo vstupu a prechod do nasledujúceho stavu.

Dve konfigurácie konečného automatu sú pax a qx , pričom platí $p, q \in Q, a \in \Sigma \cup \{\varepsilon\}, x \in \Sigma^*$ a existuje pravidlo $r \in R$ také, že $r = pa \rightarrow qa$, tak potom prechod konečného automatu

sa zapíše $pax \Rightarrow qx[r]$, alebo zjednodušene $pax \Rightarrow qx$. V literatúre sa môžeme stretnúť s tým že tento pojem prechod sa tiež nazýva derivácia konečného automatu.

Definícia 1.1.4

Sekvencia prechodov automatu M je vlastne niekoľko prechodov nasledujúcich bezprostredne za sebou.

Nech χ je konfigurácia konečného automatu M . M vykoná nula prechodov z χ do χ značíme $\chi \Rightarrow^0 \chi[\varepsilon]$, alebo zjednodušene $\chi \Rightarrow^0 \chi$.

Majme konfigurácie $\chi_0, \chi_1, \dots, \chi_n$, je to sekvencia prechodov konfigurácií pre $n \geq 1$ a $\chi_{i-1} \Rightarrow \chi_i[r_i]$ pričom platí $r_i \in R$ pre $\forall i = 1, \dots, n$ potom sa dá sekvencia prechodov zapísať: $\chi_0 \Rightarrow \chi_1[r_1] \Rightarrow \chi_2[r_2] \dots \Rightarrow \chi_n[r_n]$ a n prechodov môžeme zapísať $\chi_0 \Rightarrow^n \chi_n[r_1, \dots, r_n]$, alebo zjednodušene $\chi_0 \Rightarrow^n \chi_n$.

Potom podľa množiny do ktorej patrí n môžeme zapisovať:

- $\chi_0 \Rightarrow^* \chi_n[\rho]$ pre $n \geq 0$
- $\chi_0 \Rightarrow^+ \chi_n[\rho]$ pre $n \geq 1$

Definícia 1.1.5

Jazyk prijímaný konečným automatom M je reťazec prijímaný týmto automatom, ktorý sa z počiatočného stavu, po konečnom počte prechodov, dostane do stavu koncového:

$$L(M) = \{w : w \in \Sigma^*, sw \Rightarrow^* f, f \in F\}$$

1.2 Automatové systémy

Automatové systémy sú systémy zložené z automatov, ktoré spoločne fungujú na základe určitého protokolu, ktorý riadi ich činnosť a ich cieľom je prijať určitý jazyk. V tejto podkapitole sú ukázané systémy, ktorých komponentmi sú konečné automaty.

Automatové systémy sa v základe rozdeľujú na dve základné triedy a to na sekvenčné (Cooperation Distributed-CD) a na paralelné (Parallel Communicating-PC).

Definícia 1.2.1

Sekvenčný automatový systém je päťica symbolov:

$$M = (Q, V, \Delta, q_0, F),$$

kde Q je n -tica (Q_1, Q_2, \dots, Q_n) , kde každé Q_i je konečná množina stavov i -tej komponenty,

V je vstupná abeceda, Δ je n -tica $(\delta_1, \delta_2, \dots, \delta_n)$ prechodových funkcií, kde každá $\delta_i : Q_i \times (V \cup \{\varepsilon\}) \rightarrow 2^{\cup_i Q_i}$ a $1 \leq i \leq n$, $q_0 \in \cup_i Q_i$ je počiatočný stav a $F \subseteq \cup_i Q_i$ je množina koncových stavov.

Tieto systémy môžu pracovať vo viacerých módoch, ktoré určujú akým spôsobom si jednotlivé systémy predávajú riadenie, avšak tieto módy nemajú vplyv na silu týchto systémov, ktorá je totožná so silou konečných automatov, ako vyplýva z [3].

Definícia 1.2.2

Paralelný automatový systém je $(n + 2)$ -tica symbolov:

$$A = (V, A_1, A_2, \dots, A_n, K),$$

kde V je vstupná abeceda, $A_i = (Q_i, V_i, f_i, q_i, F_i)$, $1 \leq i \leq n$ sú konečné automaty, kde Q_i je konečná množina stavov, $q_i \in Q_i$ je počiatočný stav, V_i je konečná vstupná abeceda, f_i je prechodová funkcia definovaná $f_i : Q_i \times V \cup \{\varepsilon\} \rightarrow 2^{Q_i}$, $F \subseteq Q_i$ je množina koncových stavov a $K \subseteq \{K_1, K_2, \dots, K_n\} \subseteq \bigcup_{i=1}^n Q_i$ je množina komunikačných stavov.

Definícia 1.2.3

Konfigurácia systému A je $2n$ -tica $(s_1x_1, s_2x_2, \dots, s_nx_n)$, kde s_i je aktuálny stav i -tej komponenty a x_i je zostávajúca časť vstupného reťazca, ktorá ešte nebola spracovaná touto komponentou, pričom platí $1 \leq i \leq n$.

Existuje niekoľko druhov týchto automatových systémov, rozdelených podľa spôsobu komunikácie, respektíve podľa toho, kto a kedy môže komunikovať. Sú systémy centralizované, pri ktorých je schopná komunikácie iba jedna komponenta tzv. master, necentralizované, kde môžu komunikovať všetky komponenty, ďalej systémy pracujúce v móde s návratom, teda systémy, ktorých komponenty sa po prebehnutí komunikácie vrátia do počiatočného stavu a systémy v móde bez návratu.

Celkom existujú štyri možné varianty týchto automatových systémov:

- RCPCFA(n) – centralizované, s návratom
- RPCFA(n) – s návratom
- CPCFA(n) – centralizované
- PCFA(n) – automatové systémy.

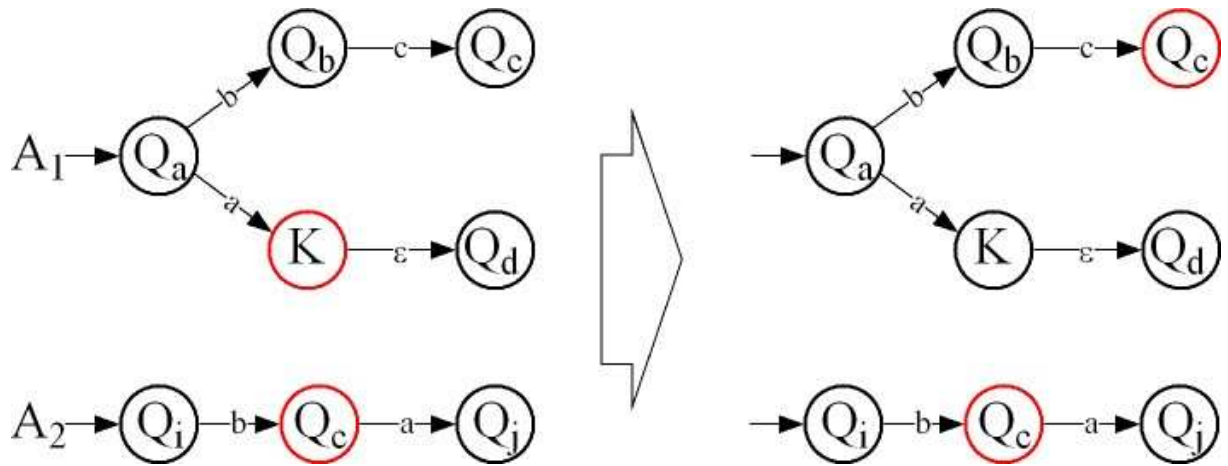
R-returning, C-centralized, Parallel Communicating Finite Automata.

Pre tieto automatové systémy platí nasledujúca definícia:

Definícia 1.2.4

Nech $X(n)$ je trieda jazykov akceptovaná automatovými systémami typu X , potom platí:

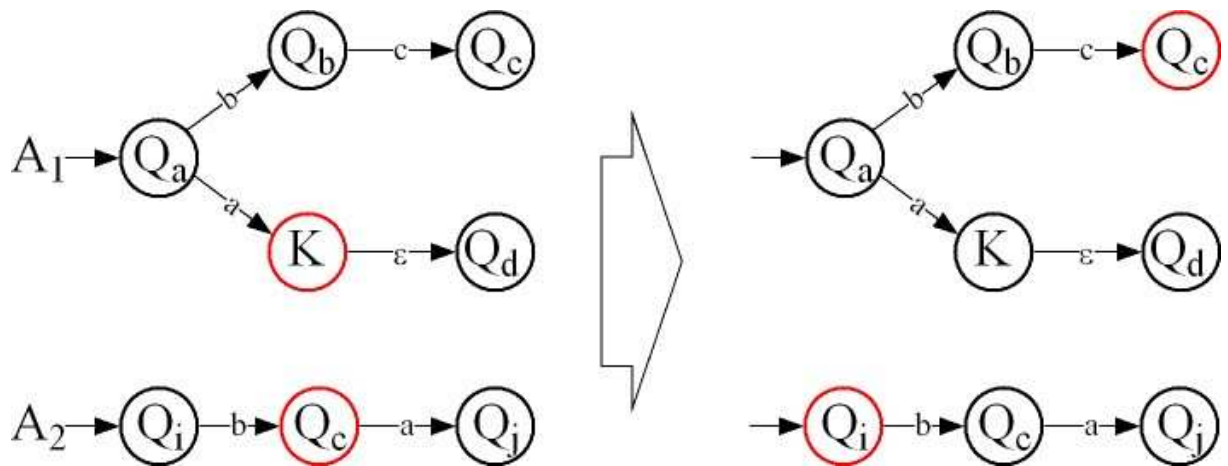
- $X(1) = REG, X \in \{RCPCFA, RPCFA, CPCFA, PCFA\}$,
- $X(n) \subseteq X(n+1)$ pre $\forall X \in \{RCPCFA, RPCFA, CPCFA, PCFA\}$



Obrázok 1.1

Nasledujúce obrázky (sú oba prevzaté z [4]) zobrazujú tieto systémy, pričom na obrázku 1.1 je zobrazený automatový systém, ktorý pracuje v móde bez návratu a je na ňom vidieť ako systém prechádza z jednej konfigurácie do nasledujúcej, pričom komunikácia má prednosť.

Nasledujúci obrázok 1.2 zobrazuje automatový systém v móde s návratom a je na ňom znázornené ako po prebehnutí komunikácie sa dotazovaná komponenta vráti do počiatočného stavu.



Obrázok 1.2

Definícia 1.2.4

Jazyk prijímaný paralelným automatovým systémom, ktorý využíva na komunikáciu stavy je definovaný:

$$L(PCFA) = \{x \in V^* \mid (q_1x, q_2x, \dots, q_nx) \Rightarrow^* (q_1\varepsilon, q_2\varepsilon, \dots, q_n\varepsilon), s_i \in F_i, 1 \leq i \leq n\}$$

Tieto automatové systémy sú podrobnejšie rozoberané v prácach [5], [3], [6].

V týchto automatových systémoch sa predpokladá, že každý automat má prístup k celému vstupnému reťazcu, avšak boli definované automatové systémy, ktoré sú založené na odlišnom princípe a to na takom, že každý automat má prístup len k časti vstupného reťazca. Tieto automaty sú definované

a študované v [7] a sú tu zmienené len niektoré odlišnosti od automatových systémov uvedených doposiaľ.

Tieto systémy sú definované nasledovne.

Definícia 1.2.5

Paralelný komunikujúci automatový systém komunikujúci prechodmi je $(n+1)$ -tica symbolov:

$$AS = (\Sigma, A_1, \dots, A_n),$$

kde Σ je vstupná abeceda $A_i = (Q_i, \Sigma_i, R_i, q_i, F_i)$, $1 \leq i \leq n$ sú konečné automaty, kde Q_i je konečná množina stavov, $q_i \in Q_i$ je počiatočný stav, $\Sigma_i = \Sigma$ je vstupná abeceda,

$$R_i = \{pa \rightarrow q \mid p, q \in Q_i, a \in \Sigma_i \cup \{\varepsilon\}\} \cup \left\{ p \xrightarrow{c} q \mid p, q \in Q_i, c \in \bigcup_{j=1}^n Q_j \right\}$$
 je prechodová funkcia,

ktorá obsahuje klasické pravidlá $pa \rightarrow q$ a zároveň aj komunikačné pravidlá $pa \xrightarrow{c} q$, $F \subseteq Q_i$

je množina koncových stavov a platí $\bigcap_{j=1}^n Q_j = \emptyset$.

Tieto systémy sú taktiež založené na paralelnom princípe, avšak komunikácia prebieha pomocou komunikačných pravidiel, pričom vstupný reťazec je úspešne prijatý ak sa prvá komponenta (tzv. master) dostane do koncového stavu a celý reťazec je prijatý, pričom nezáleží na tom, v akom stave skončia ostatné komponenty.

Takéto systémy nemajú žiadne obmedzenie týkajúce sa komunikácie a teda každá komponenta môže obsahovať komunikačné stavy, pomocou ktorých môže komunikovať s ostatnými systémami.

1.3 Kanonické multigeneratívne gramatické systémy

V tejto podkapitole sú ukázané kanonické gramatické systémy a popísané niektoré ich základné vlastnosti, tieto systémy sú definované a podrobne popísane v [8] a pre podrobnejšie informácie o oblasti gramatík a gramatických systémov je vhodné si prečítať práce [1], [9].

Tieto systémy generujú reťazce podľa najľavšej derivácie, teda v každom kroku sa v danej vetnej forme prepíše vždy ten symbol, ktorý je najviac vľavo. Obecne sa tieto gramatické systémy skladajú z n -bezkontextových gramatík a jeden prechod je vykonaný tak, že sa paralelne u všetkých n gramatík aplikuje príslušné pravidlo na aktuálnu vetnú formu a tieto prechody sú kontrolované n -ticou nonterminálov, alebo n -ticou pravidiel.

Sú definované dva základné typy týchto systémov a to kanonický n -generatívny nonterminálovo synchronizovaný gramatický systém a kanonický n -generatívny pravidlovo synchronizovaný gramatický systém.

1.3.1 Kanonický n -generatívny nonterminálovo synchronizovaný gramatický systém

Kanonický n -generatívny nonterminálovo synchronizovaný gramatický systém je zložený z n bezkontextových gramatík a zo špeciálnej kontrolnej komponenty Q , ktorá je množinou n -tíc nonterminálových systémov. Derivačný krok je zložený z dvoch častí, v prvej časti sa vykoná kontrola vetných foriem vygenerovaných jednotlivými gramatikami a ak je n -tica, ktorá pri kontrole vznikne obsiahnutá v kontrolnej komponente Q , tak sa pokračuje druhou časťou a inak je derivácia zablokovaná. V druhej časti sa paralelne v každej gramatike aplikuje práve jedno pravidlo v najľavšej derivácii na aktuálnu vetnú formu.

Definícia 1.3.1

Kanonický n -generatívny nonterminálovo synchronizovaný GS (n -KGN) je $n + 1$ -tica symbolov:

$$\Gamma = (G_1, G_2, \dots, G_n, Q),$$

kde, $G_i = (N_i, T_i, P_i, S_i)$ je bezkontextová gramatika pre všetky $1 \leq i \leq n$ a Q je konečná množina kontrolných n -tíc nonterminálov tvaru $Q = (A_1, A_2, \dots, A_n)$, $A_i \in N_i$ pre všetky $i = 1, \dots, n$.

1.3.2 Kanonický n -generatívny pravidlovo synchronizovaný gramatický systém

Kanonický n -generatívny pravidlovo synchronizovaný gramatický systém sa skladá z n bezkontextových gramatík a zo špeciálnej kontrolnej komponenty Q , ktorá je množinou n -tíc pravidiel. Derivačný krok sa vykoná tak, že z komponenty Q sa vyberie jedna n -tica pravidiel, pričom prvé pravidlo z tejto n -tice je aplikované na vetnú formu vygenerovanú prvou gramatikou, druhé pravidlo na vetnú formu vygenerovanú druhou gramatikou atď. Všetky tieto pravidlá sú aplikované na najľavší nonterminálny symbol v jednotlivých vetných formách.

Definícia 1.3.2

Kanonický n -generatívny pravidlovo synchronizovaný GS (n -KGN) je $n + 1$ -tica symbolov:

$$\Gamma = (G_1, G_2, \dots, G_n, Q),$$

kde $G_i = (N_i, T_i, P_i, S_i)$ je bezkontextová gramatika pre všetky $1 \leq i \leq n$ a Q je konečná množina kontrolných n -tíc pravidiel tvaru $Q = (p_1, p_2, \dots, p_n)$, $p_i \in P_i$ pre všetky $1 \leq i \leq n$.

2 Definovanie nových automatových systémov

V tejto kapitole sú definované dva nové typy automatových systémov, ktorých komponentmi sú konečné automaty.

Tieto systémy sú založené na paralelnom spracovaní, pričom prijímaný reťazec je rozdelený na viacero častí a každá komponenta systému spracováva určitú časť reťazca. Činnosť je riadená na základe kontroly buď stavov, alebo pravidiel. Stavov, v ktorých sa nachádzajú jednotlivé komponenty systému po vykonanom prechode do nasledujúcej konfigurácie, alebo pravidiel, ktoré sú pri prechode použité.

Pri návrhu systémov boli využité princípy z definícií kanonických gramatických systémov, pričom bola snaha tieto princípy činnosti aplikovať do oblasti automatových systémov. Tieto systémy sú v základoch popísané v kapitole 1.3 a podrobne sú študované v [8] na stranách 20 až 37.

2.1 Stavovo kontrolovaný paralelný automatový systém

Tento systém je zložený zo vstupnej abecedy Σ , n konečných automatov a kontrolnej komponenty, ktorá obsahuje kontrolné n -tice, ktoré určujú, či je uskutočnený prechod správny, na základe kontroly stavov, do ktorých sa tento systém po prechode dostal. Ak stavy systému a stavy v kontrolnej n -tici súhlasia, tak sa pokračuje v spracovávaní vstupného reťazca, inak je spracovávanie zastavené a môžeme potom tvrdiť že systém daný jazyk neprijíma.

Definícia 2.1.1

Stavovo kontrolovaný paralelný automatový systém (n -SKP) je $(n + 2)$ -tica symbolov:

$$T = (\Sigma, A_1, \dots, A_n, C),$$

kde Σ je vstupná abeceda, $A_i = (Q_i, \Sigma_i, R_i, q_i, F_i)$, $1 \leq i \leq n$ sú konečné automaty a C je konečná množina kontrolných n -tíc stavov jednotlivých komponent v tvare $C = (c_1, c_2, \dots, c_n)$, $c_i = \{q_1, \dots, q_n\}$, $q_i \in Q_i$, pričom $1 \leq i \leq n$.

Definícia 2.1.2

Nech $\chi_1 = q_1 v_1 q_2 v_2 \dots q_n v_n$ a $\chi_{1*} = q_{1*} v_{1*} q_{2*} v_{2*} \dots q_{n*} v_{n*}$ sú dve konfigurácie n -SKP, pričom $q_i, q_{i*} \in Q_i$, $v_i, v_{i*} \in \Sigma^*$, $1 \leq i \leq n$ a pre každé $i \in \{1, \dots, n\}$ $\exists r_i = (q_i a_i \rightarrow q_{i*}) \in R_i$, pričom $a_i \in \Sigma \cup \{\varepsilon\}$, $v_i = a_i v_{i*}$ a zároveň $\exists c_i = \{q_1, q_2, \dots, q_n\} \in C$, $1 \leq i \leq n$, tak potom sa môže

uskutočniť prechod z konfigurácie χ_1 do konfigurácie χ_2 za použitia pravidiel r_1, r_2, \dots, r_n , a môže byť použitý nasledovný zápis $\chi_1 \Rightarrow \chi_2[r_1, r_2, \dots, r_n]$, alebo zjednodušene $\chi_1 \Rightarrow \chi_2$.

Definícia 2.1.3

Sekvencia prechodov n -SKP je definovaná:

ak χ je konfigurácia tak potom sa nula prechodov z χ do χ zapíše $\chi \Rightarrow^0 \chi$.

Nech $\chi_0, \chi_1, \dots, \chi_n$ sú konfigurácie pri ktorých pre všetky $i = 1, \dots, n$ platí: $\chi_{i-1} \Rightarrow \chi_i$ potom môžeme povedať, že z konfigurácie χ_0 sa po n krokoch dostane systém do konfigurácie χ_n a zapíše sa to nasledovne $\chi_0 \Rightarrow^n \chi_n$.

Potom ak $\chi_0 \Rightarrow^n \chi_n$ tak podľa množiny, do ktorej patrí n sa môže zapísať:

- $\chi_0 \Rightarrow^* \chi_n[\rho]$ pre $n \geq 0$
- $\chi_0 \Rightarrow^+ \chi_n[\rho]$ pre $n \geq 1$

Definícia 2.1.4

Jazyk prijímaný n -SKP automatovým systémom je definovaný:

$$L(T) = \{w | q_1 v_1 q_2 v_2 \dots q_n v_n \Rightarrow^* f_1, f_2, \dots, f_n, w \in \Sigma^*, v_1 v_2 \dots v_n = w, f_i \in F_i, i \in \{1, \dots, n\}\}$$

Príklad 2.1

Máme 2-SKP $T = (\Sigma, A_1, A_2, C)$ kde:

- $\Sigma = \{a, b, c\}$,
- $A_1 = (\{S_{11}, S_{12}, S_{13}, F_1\}, \{a, b, c\}, R_1, S_{11}, F_1)$,
 $R_1 = \{S_{11}a \rightarrow S_{12}, S_{12}b \rightarrow S_{12}, S_{12}c \rightarrow S_{13}, S_{13}a \rightarrow F_1\}$
- $A_2 = (\{S_{21}, S_{22}, S_{23}, S_{24}\}, \{a, b, c\}, R_2, S_{21}, \emptyset)$,
 $R_2 = \{S_{21}b \rightarrow S_{22}, S_{22}c \rightarrow S_{22}, S_{22}a \rightarrow S_{23}, S_{23}b \rightarrow S_{24}\}$
- $C = \{(S_{11}, S_{21}), (S_{12}, S_{22}), (S_{12}, S_{22}), (S_{12}, S_{22}), (S_{13}, S_{23}), (F_1, S_{24})\}$.

Jedná sa o stavovo kontrolovaný paralelný automatový systém, ktorý má dve komponenty, ktorými sú konečné automaty.

Jazyk prijímaný týmto automatovým systémom je nasledovný:

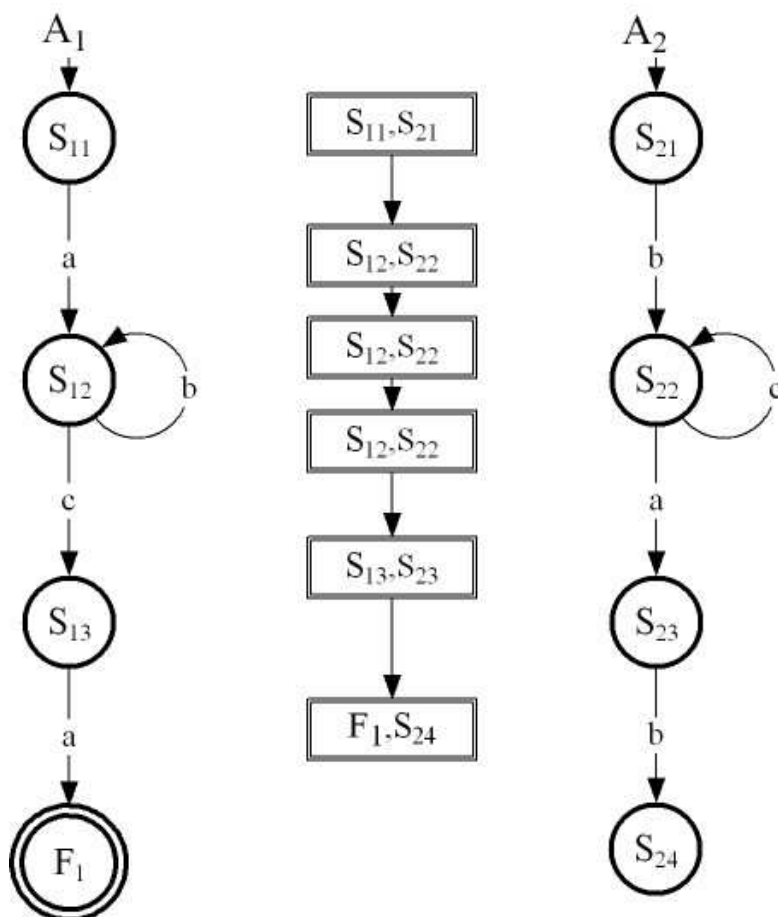
$$L(T) = \{ab^n cab c^n ab | n \geq 0\}$$

Tento systém spracuje vstupný reťazec $abbcabccab$ nasledujúcim spôsobom:

1. $S_{11}abbcas_{21}bccab$
2. $S_{12}bbcas_{22}ccab$
3. $S_{12}bcas_{22}cab$

4. $S_{12}caS_{22}ab$
5. $S_{13}aS_{23}b$
6. F_1S_{24}

Na obrázku 2.1 je celá situácia znázornená:



Obrázok 2.1

2.2 Pravidlovo kontrolovaný paralelný automatový systém

Tento systém je zložený zo vstupnej abecedy Σ , n konečných automatov a kontrolnej komponenty C , ktorá obsahuje kontrolné n -tice, ktoré určujú, či je uskutočnený prechod správny, na základe kontroly pravidiel, ktoré boli použité pri prechode do ďalšieho stavu. Ak pravidlá aplikované pri prechode súhlasia s pravidlami v kontrolnej n -tici tak sa pokračuje v spracovávaní vstupného reťazca, inak je spracovávanie zastavené a môžeme tvrdiť že systém daný jazyk neprijíma.

Definícia 2.2.1

Pravidlovo kontrolovaný paralelný automatový systém (n -PKP) je $n + 2$ -tica symbolov:

$$T = (\Sigma, A_1, \dots, A_n, C),$$

kde Σ je vstupná abeceda, $A_i = (Q_i, \Sigma_i, R_i, q_i, F_i)$, $1 \leq i \leq n$ sú konečné automaty a C je konečná množina kontrolných n -tíc pravidiel jednotlivých komponent v tvare $C = (c_1, c_2, \dots, c_n)$, $c_i = \{r_1, \dots, r_n\}$, $r_i \in R_i$, pričom $1 \leq i \leq n$.

Definícia 2.2.2

Nech $\chi_1 = q_1 v_1 q_2 v_2 \dots q_n v_n$ a $\chi_{1*} = q_{1*} v_{1*} q_{2*} v_{2*} \dots q_{n*} v_{n*}$ sú dve konfigurácie n -PKP, pričom $q_i, q_{i*} \in Q_i$, $v_i, v_{i*} \in \Sigma^*$, $1 \leq i \leq n$ a pre každé $i \in \{1, \dots, n\}$ $\exists r_i = (q_i a_i \rightarrow q_{i*}) \in R_i$, pričom $a_i \in \Sigma \cup \{\varepsilon\}$, $v_i = a_i v_{i*}$ a zároveň $\exists c_i = \{r_1, r_2, \dots, r_n\} \in C$, $1 \leq i \leq n$ tak potom sa môže uskutočniť prechod z konfigurácie χ_1 do konfigurácie χ_2 za použitia pravidiel r_1, r_2, \dots, r_n , a zapíš sa to nasledovne $\chi_1 \Rightarrow \chi_2[r_1, r_2, \dots, r_n]$, alebo zjednodušene $\chi_1 \Rightarrow \chi_2$.

Definícia 2.2.3

Sekvencia prechodov n -PKP je definovaná obdobne ako sekvencia prechodov u n -SKP.

Definícia 2.2.4

Jazyk prijímaný n -PKP automatovým systémom je definovaný obdobne ako pri n -SKP.

Príklad 2.2

Máme 2-PKP $T = (\Sigma, A_1, A_2, C)$ kde:

- $\Sigma = \{a, b, c\}$,
- $A_1 = (\{S_{11}, S_{12}, S_{13}, F_1\}, \{a, b, c\}, R_1, S_{11}, F_1)$,
 $R_1 = \{1 : S_{11}a \rightarrow S_{12}, 2 : S_{12}b \rightarrow S_{13}, 3 : S_{13}c \rightarrow S_{13}, 4 : S_{13}a \rightarrow F_1\}$
- $A_2 = (\{S_{21}, S_{22}, S_{23}, S_{24}\}, \{a, b, c\}, R_2, S_{21}, \emptyset)$,
 $R_2 = \{1 : S_{21}b \rightarrow S_{22}, 2 : S_{22}c \rightarrow S_{23}, 3 : S_{23}a \rightarrow S_{23}, 4 : S_{23}b \rightarrow S_{24}\}$
- $C = \{(1,1), (2,2), (3,3), (3,3), (4,4)\}$.

Jedná sa o stavovo kontrolovaný paralelný automatový systém, ktorý má dve komponenty, ktorými sú konečné automaty.

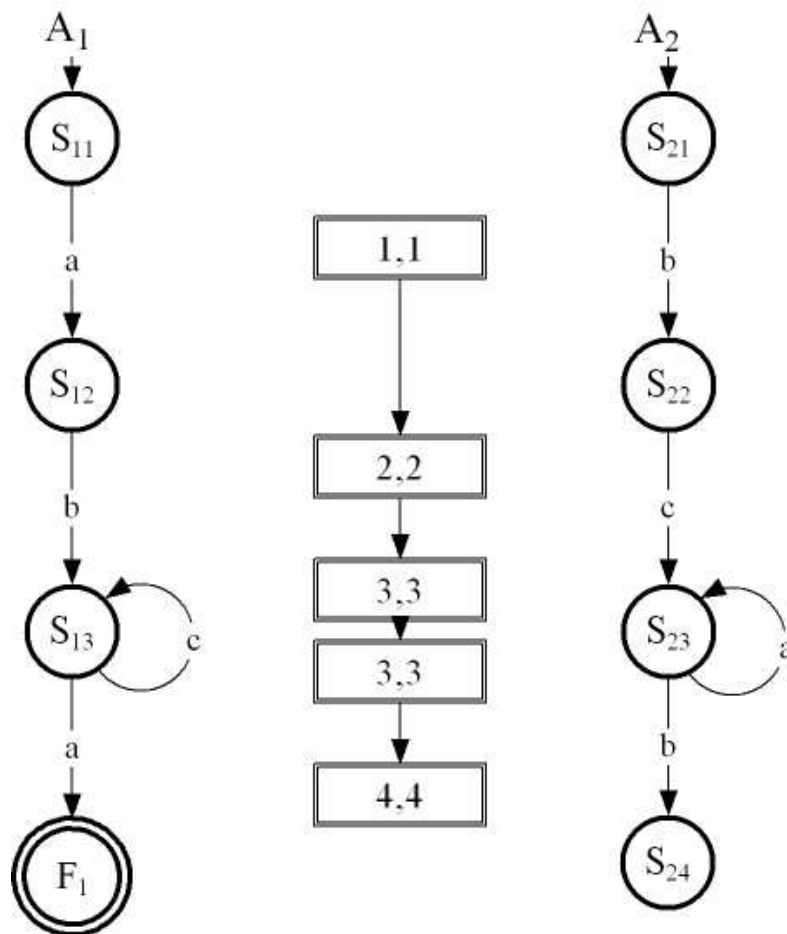
Jazyk prijímaný týmto automatovým systémom je nasledovný:

$$L(T) = \{abc^n abca^n b \mid n \geq 0\}$$

Tento systém spracuje vstupný reťazec *abccabcaab* nasledovným spôsobom:

1. $S_{11}abccaS_{21}bcaab$
2. $S_{12}bccaS_{22}caab$
3. $S_{12}ccaS_{22}aab$
4. $S_{12}caS_{22}ab$
5. $S_{13}aS_{23}b$
6. F_1S_{24}

Na obrázku 2.2 je celá situácia znázornená.



Obrázok 2.2

2.3 Vzťah medzi n-SKP a n-PKP

Stavovo a aj pravidlovo kontrolované paralelné automatové systémy sú oba schopné prijať ten istý jazyk a teda oba tieto automatové systémy majú pretože tú istú silu a na základe tohto predpokladu je

možné uvažovať o algoritmoch, podľa ktorých by malo byť možné previesť jeden automatový systém na druhý a naopak a tieto algoritmy sú v nasledujúcich podkapitolách popísané.

2.3.1 Prevod n -SKP na ekvivalentný n -PKP

Nasledujúci algoritmus popisuje, ako previesť ľubovoľný n -SKP na ekvivalentný n -PKP, teda taký, ktorý dokáže prijať ten istý jazyk.

V n -SKP sú kontrolované n -tice stavov jednotlivých komponent, zatiaľ čo v n -PKP budú kontrolované n -tice pravidiel a teda vstupná abeceda a aj jednotlivé komponenty zostávajú rovnaké a bude sa jednať hlavne o to, ako správne pretransformovať kontrolnú množinu stavov na kontrolnú množinu pravidiel. Pretože ak bola v pôvodnej kontrolnej komponente C napríklad n -tica $(Q_{12}, Q_{22}, \dots, Q_{n2})$, tak potom v novej kontrolnej komponente C_{new} , musia byť obsiahnuté také n -tice pravidiel, ktoré majú na ľavej strane práve pôvodné prvky kontrolnej komponenty a teda napríklad $(Q_{11}a \rightarrow Q_{12}, Q_{21}b \rightarrow Q_{22}, \dots, Q_{n1}x \rightarrow Q_{n2})$. Ďalšou podmienkou pri správnej transformácii je prihliadnutie k predchádzajúcej a aj k nasledujúcej kontrolnej n -tici stavov a na základe tejto informácie vybrať správne pravidlo, táto podmienka rieši situáciu, kedy je pri výbere pravidla viacero možností a teda je to napríklad taká situácia, keď množina pravidiel obsahuje napríklad pravidlá $1: S_{12}b \rightarrow S_{12}, 2: S_{12}c \rightarrow S_{13}$ a kontrolná n -tica stavov obsahuje napríklad tieto prvky $(S_{12}, S_{22}), (S_{12}, S_{22}), (S_{12}, S_{22}), (S_{13}, S_{23})$, tak potom správny výber je nasledovné poradie pravidiel **1, 1, 2**.

Ďalšia podmienka sa uplatňuje v tom prípade, že kontrolná n -tica stavov obsahuje stav, ktorý je pre jednu z komponent konečný a tak v tomto prípade sa táto kontrolná n -tica už netransformuje.

Takto je zaručené, že sa tá správna kontrolná n -tica pravidiel vyberie práve v tej istej situácii v akej by bola vybraná pôvodná kontrolná n -tica stavov.

Algoritmus 2.1

Vstupom je n -SKP $T = (\Sigma, A_1, \dots, A_n, C)$ a výstupom je n -PKP $T_{new} = (\Sigma, A_1, \dots, A_n, C_{new})$, pričom platí, $L(T) = L(T_{new})$.

Princíp prevodu je nasledovný:

Ak $A_i = (Q_i, \Sigma_i, R_i, q_i, F_i)$, $1 \leq i \leq n$ tak potom:

$C_{new} = (\{Q_{11}x_1 \rightarrow Q_{12}, Q_{21}x_2 \rightarrow Q_{22}, \dots, Q_{n1}x_n \rightarrow Q_{n2}\}, Q_{i1}x_i \rightarrow Q_{i2}, (Q_{11}, Q_{12}, Q_{21}, \dots, Q_{n1}) \in Q$, pre všetky $1 \leq i \leq n$.

Príklad 2.3

Máme nasledujúci 2-SKP $T = (\Sigma, A_1, A_2, C)$ kde:

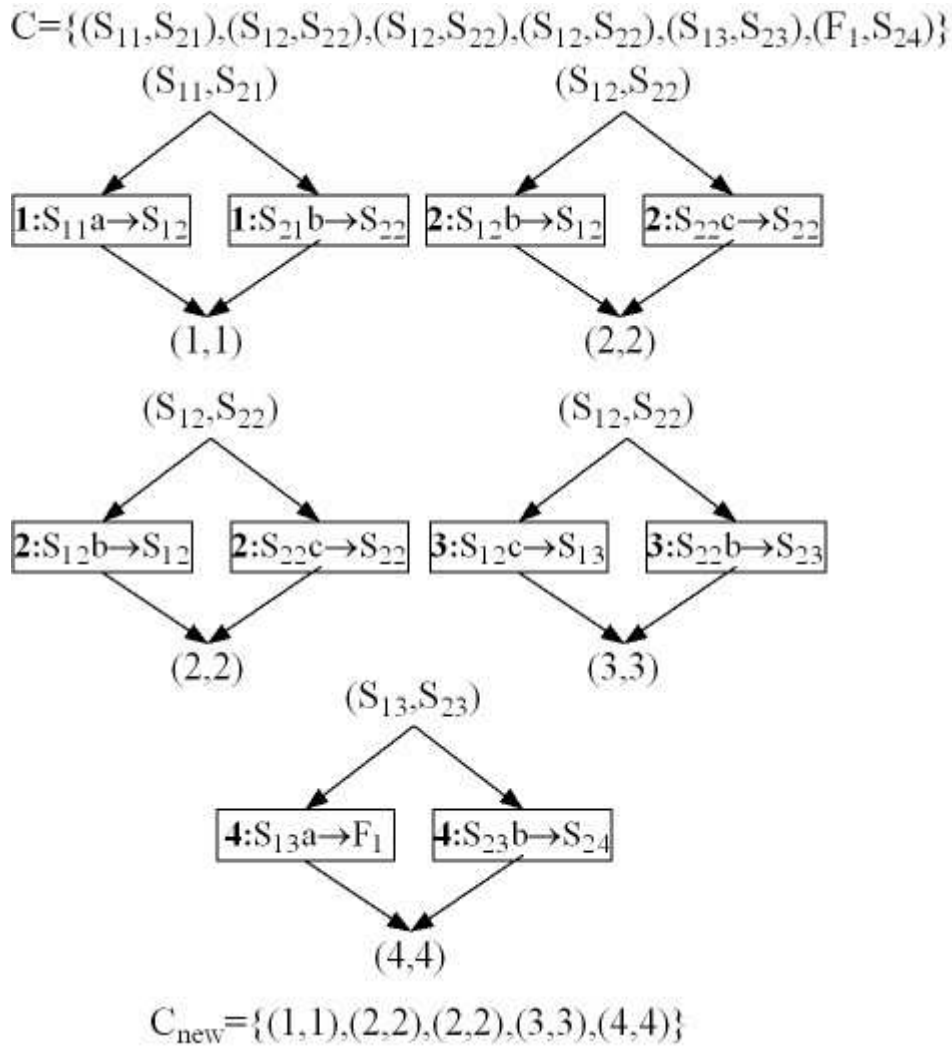
- $\Sigma = \{a, b, c\}$,
- $A_1 = (\{S_{11}, S_{12}, S_{13}, F_1\}, \{a, b, c\}, R_1, S_{11}, F_1)$,
 $R_1 = \{S_{11}a \rightarrow S_{12}, S_{12}b \rightarrow S_{12}, S_{12}c \rightarrow S_{13}, S_{13}a \rightarrow F_1\}$

- $A_2 = (\{S_{21}, S_{22}, S_{23}, S_{24}\}, \{a, b, c\}, R_2, S_{21}, \emptyset)$,
 $R_2 = \{S_{21}b \rightarrow S_{22}, S_{22}c \rightarrow S_{22}, S_{22}a \rightarrow S_{23}, S_{23}b \rightarrow S_{24}\}$
- $C = \{(S_{11}, S_{21}), (S_{12}, S_{22}), (S_{12}, S_{22}), (S_{12}, S_{22}), (S_{13}, S_{23}), (F_1, S_{24})\}$.

Po použití algoritmu 2.1 z tohoto 2-SKP automatového systému vznikne ekvivalentný automatový systém 2-PKP $T_{new} = (\Sigma, A_1, A_2, C_{new})$, ktorý je definovaný nasledovne:

- $\Sigma = \{a, b, c\}$,
- $A_1 = (\{S_{11}, S_{12}, S_{13}, F_1\}, \{a, b, c\}, R_1, S_{11}, F_1)$,
 $R_1 = \{1 : S_{11}a \rightarrow S_{12}, 2 : S_{12}b \rightarrow S_{12}, 3 : S_{12}c \rightarrow S_{13}, 4 : S_{13}a \rightarrow F_1\}$
- $A_2 = (\{S_{21}, S_{22}, S_{23}, S_{24}\}, \{a, b, c\}, R_2, S_{21}, \emptyset)$,
 $R_2 = \{1 : S_{21}b \rightarrow S_{22}, 2 : S_{22}c \rightarrow S_{22}, 3 : S_{22}a \rightarrow S_{23}, 4 : S_{23}b \rightarrow S_{24}\}$
- $C_{new} = \{(1,1), (2,2), (2,2), (3,3), (4,4)\}$.

Postup vytváranie toho automatového systému je ukázaný na obrázku 2.1:



Obrázok 2.1

2.3.2 Prevod n-PKP na ekvivalentný n-SKP

Nasledujúci algoritmus popisuje, ako previesť ľubovoľný n -PKP na ekvivalentný n -SKP, teda taký, ktorý dokáže prijať ten istý jazyk.

V n -PKP sú kontrolované n -tice pravidiel, zatiaľ čo v n -SKP bude kontrolovaná n -tica stavov príslušných komponent a teda vstupná abeceda a aj jednotlivé komponenty zostávajú rovnaké a bude sa jednať hlavne o to, ako správne pretransformovať kontrolnú množinu pravidiel na kontrolnú množinu stavov. Pretože ak bola v pôvodnej kontrolnej komponente C napríklad n -tica $(Q_{11}a \rightarrow Q_{12}, Q_{21}b \rightarrow Q_{22}, \dots, Q_{n1}x \rightarrow Q_{n2})$, tak potom v novej kontrolnej komponente C_{new} , musia byť obsiahnuté také n -tice stavov, ktoré sú na pravej strane príslušného pravidla v pôvodnej kontrolnej komponente a teda vznikne napríklad n -tica $(Q_{12}, Q_{22}, \dots, Q_{n2})$. Toto všetko platí pre vytváranie všetkých kontrolných n -tíc, okrem tej úplne prvej, ktorá obsahuje vždy tie stavy komponent, ktoré sú pre jednotlivé komponenty počiatočné.

Takto je potom zaručené, že sa tá správna kontrolná n -tica stavov, použije práve v prípade, kedy by sa pred transformáciou použila kontrolná n -tica pravidiel.

Algoritmus 2.2

Vstupom je n -SKP $T = (\Sigma, A_1, \dots, A_n, C)$ a výstupom je n -PKP $T_{new} = (\Sigma, A_1, \dots, A_n, C_{new})$, pričom platí, $L(T) = L(T_{new})$.

Princíp prevodu je nasledovný:

Ak $A_i = (Q_i, \Sigma_i, R_i, q_i, F_i)$, $1 \leq i \leq n$ tak potom:
 $C_{new} = (\{Q_{11}x_1 \rightarrow Q_{12}, Q_{21}x_2 \rightarrow Q_{22}, \dots, Q_{n1}x_n \rightarrow Q_{n2}\}, Q_{i1}x_i \rightarrow Q_{i2}, (Q_{11}, Q_{12}, Q_{21}, \dots, Q_{n1}) \in Q$
, pre všetky $1 \leq i \leq n$.

Príklad 2.4

Máme nasledujúci 2-PKP $T = (\Sigma, A_1, A_2, C)$ kde:

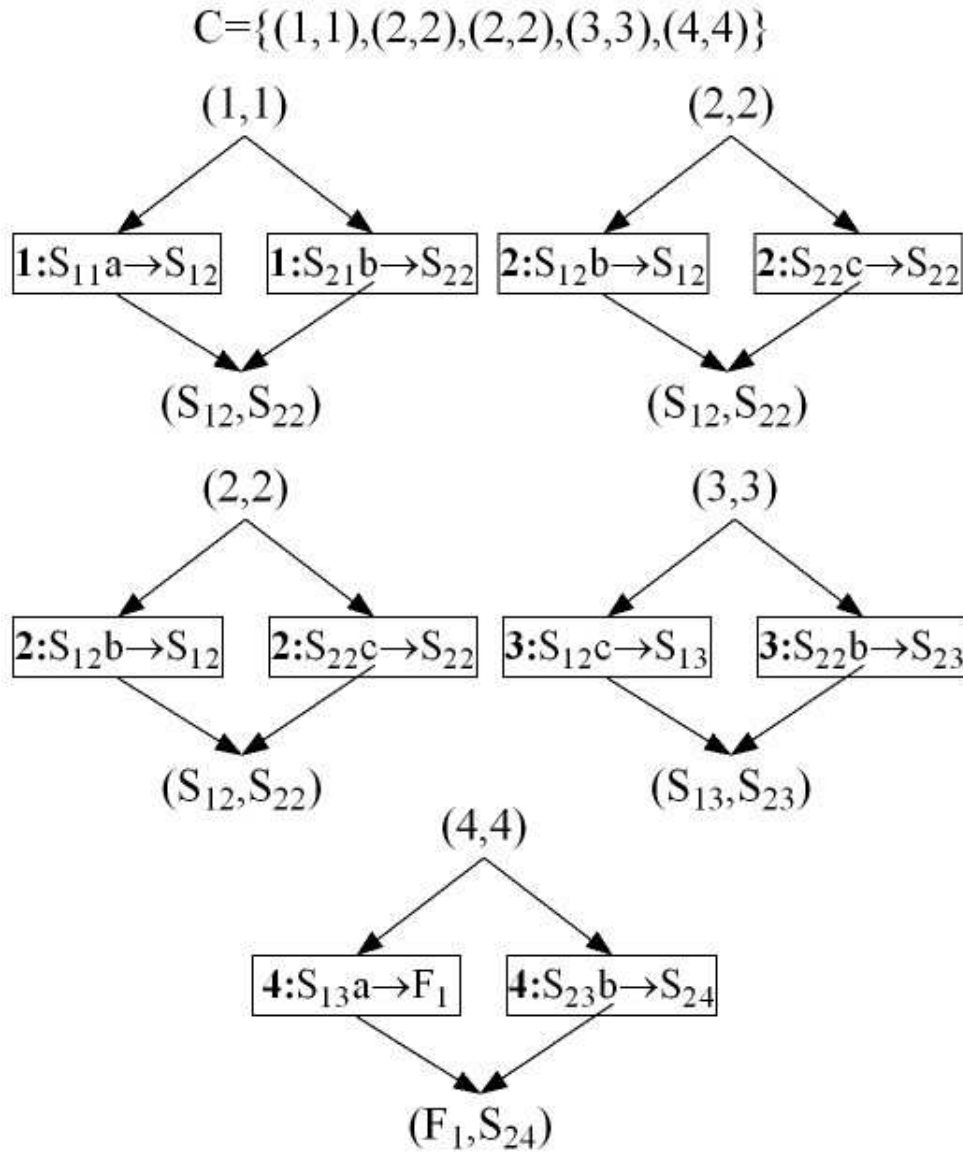
- $\Sigma = \{a, b, c\}$,
- $A_1 = (\{S_{11}, S_{12}, S_{13}, F_1\}, \{a, b, c\}, R_1, S_{11}, F_1)$,
 $R_1 = \{1 : S_{11}a \rightarrow S_{12}, 2 : S_{12}b \rightarrow S_{12}, 3 : S_{12}c \rightarrow S_{13}, 4 : S_{13}a \rightarrow F_1\}$
- $A_2 = (\{S_{21}, S_{22}, S_{23}, S_{24}\}, \{a, b, c\}, R_2, S_{21}, \emptyset)$,
 $R_2 = \{1 : S_{21}b \rightarrow S_{22}, 2 : S_{22}c \rightarrow S_{22}, 3 : S_{22}a \rightarrow S_{23}, 4 : S_{23}b \rightarrow S_{24}\}$
- $C = \{(1,1), (2,2), (3,3), (4,4)\}$.

Po použití algoritmu 2.2 z tohoto 2-PKP automatového systému vznikne ekvivalentný automatový systém 2-SKP $T_{new} = (\Sigma, A_1, A_2, C_{new})$, ktorý je definovaný nasledovne:

- $\Sigma = \{a, b, c\}$,
- $A_1 = (\{S_{11}, S_{12}, S_{13}, F_1\}, \{a, b, c\}, R_1, S_{11}, F_1)$,
 $R_1 = \{S_{11}a \rightarrow S_{12}, S_{12}b \rightarrow S_{12}, S_{12}c \rightarrow S_{13}, S_{13}a \rightarrow F_1\}$

- $A_2 = (\{S_{21}, S_{22}, S_{23}, S_{24}\}, \{a, b, c\}, R_2, S_{21}, \emptyset)$,
 $R_2 = \{S_{21}b \rightarrow S_{22}, S_{22}c \rightarrow S_{22}, S_{22}a \rightarrow S_{23}, S_{23}b \rightarrow S_{24}\}$
- $C_{new} = \{(S_{11}, S_{21}), (S_{12}, S_{22}), (S_{12}, S_{22}), (S_{12}, S_{22}), (S_{13}, S_{23}), (F_1, S_{24})\}$.

Postup vytváranie toho automatového systému je ukázaný na obrázku 2.2:



Obrázok 2.2

2.4 Vlastnosti n -SKP a n -PKP a porovnanie s ostatnými automatovými systémami

V tejto kapitole boli definované dva nové typy automatových systémov, konkrétne stavovo kontrolovaný paralelný automatový systém (n -SKP) a pravidlovo kontrolovaný paralelný automatový systém (n -PKP) a v tejto podkapitole sú zhrnuté a skúmané niektoré ich vlastnosti a v závere je ich ukážkové porovnanie s paralelne komunikujúcimi automatovými systémami, komunikujúcich prechodmi z kapitoly 1.

Oproti automatovým systémom, ktorých základné rysy boli popísané v kapitole 1.2 pracujú stavovo a pravidlovo kontrolované paralelné automatové systémy na odlišnom princípe. Zatiaľ čo systémy z 1.2 využívajú ako prostriedok pri riadení svojej činnosti komunikáciu a to či už prostredníctvom komunikačných stavov, alebo komunikačných pravidiel, tak činnosť n -SKP a n -PKP je riadená na základe kontrolnej komponenty C , ktorá určí, či je vykonaný derivačný krok správny, alebo nie a to na základe príslušnej n -tice obsiahnutej v tejto komponente C , ktorá obsahuje buď príslušné kontrolné stavy, do ktorých sa má po prechode systém dostať, alebo príslušné kontrolné pravidlá, ktoré majú byť pri prechode použité.

Ďalšou vlastnosťou, ktorou sa odlišujú od systémov z kapitoly 1.2 je to, že pri týchto systémoch sa nedá hovoriť o riadení činnosti v zmysle tom, že jedna komponenta by mala riadiť celý systém a teda mať funkciu mastra, zatiaľ čo pri systémoch z 1.2 je pri komunikácii viacero možností ako riadiť komunikáciu, ktorú komponentu označiť za mastra a podobne. Tak isto sa pri týchto systémoch nevyskytujú ďalšie problémy súvisiace s komunikáciou, ako napríklad možnosť zaseknutia sa systému v prípade, že jednotlivé komponenty budú v kruhu čakať na to, kým nasledujúce komponenty uskutočnia komunikáciu.

Jednou z vlastností, ktorá je pre systémy n -SKP, n -PKP a paralelne komunikujúce systémy komunikujúce prechodmi spoločná je spôsob, akým prijímajú reťazec, ten sa neprijíma celý ako pri ostatných systémoch z kapitoly 1, ale je rozdelený na viacero častí a každá komponenta spracováva svoju časť reťazca a tak isto je to aj vlastnosť, že pre prijatie určitého reťazca nie je podstatné to, aby sa každá komponenta dostala do koncového stavu, ale podstatné je to aby bol reťazec spracovaný.

Nasledujúci príklad demonštruje, ako automatový systém z kapitoly 2 a paralelne komunikujúci automatový systém komunikujúci prechodmi prijímajú ten istý reťazec.

Príklad 2.5

Majme dva automatové systémy, jeden je stavovo kontrolovaný paralelný automatový systém definovaný nasledovne:

$$2\text{-SKP } T = (\Sigma, A_1, A_2, C) \text{ kde:}$$

- $\Sigma = \{a, b, c\}$,
- $A_1 = (\{S_{11}, S_{12}, S_{13}\}, \{a, b, c\}, R_1, S_{11}, \emptyset)$,
 $R_1 = \{S_{11}b \rightarrow S_{11}, S_{11}c \rightarrow S_{12}, S_{12}c \rightarrow S_{13}, S_{13}c \rightarrow S_{12}\}$
- $A_2 = (\{S_{21}, S_{22}, S_{23}\}, \{a, b, c\}, R_2, S_{21}, \{S_{23}\})$,
 $R_2 = \{S_{21}a \rightarrow S_{22}, S_{22}a \rightarrow S_{21}, S_{21}b \rightarrow S_{23}, S_{23}b \rightarrow S_{23}\}$

- $C = \{(S_{11}, S_{21}), (S_{11}, S_{22}), (S_{12}, S_{21}), (S_{13}, S_{23}), (S_{12}, S_{23})\}$.

a druhým je paralelný centralizovaný automatový systém komunikujúci prechodmi definovaný takto:

$$\text{CPC } AS = (\{a, b, c\}, A_1, A_2) \text{ kde:}$$

- $A_1 = (\{Q_1, Q_2, Q_3, Q_4, Q_5, Q_6, Q_7\}, \{a, b, c\}, R_1, Q_1, \{Q_5\})$,

$$R_1 = \left\{ \begin{array}{l} Q_1 \varepsilon \rightarrow Q_2, Q_2 \varepsilon \rightarrow Q_3, Q_3 \varepsilon \rightarrow Q_5, Q_5 c \rightarrow Q_6, Q_6 c \rightarrow Q_7, Q_4 b \rightarrow Q_1, \\ Q_3 \xrightarrow{A} Q_4, Q_7 \xrightarrow{B} Q_5 \end{array} \right\}$$
- $A_2 = (\{P_1, P_2, P_3, P_4, P_5, P_6, A, B\}, \{a, b, c\}, R_2, P_1, \emptyset)$,

$$R_2 = R_1 = \left\{ \begin{array}{l} P_1 a \rightarrow P_2, P_2 a \rightarrow A, A \varepsilon \rightarrow P_3, P_3 \varepsilon \rightarrow P_1, P_1 b \rightarrow P_4, P_4 \varepsilon \rightarrow B, \\ B \varepsilon \rightarrow P_6, P_6 b \rightarrow B \end{array} \right\}.$$

a majme na vstupe reťazec, ktorý chceme prijať $bccaabb$ a teda jazyk, ktorý by mali byť systémy schopné prijať je $L = \{b^n c^{2m} a^{2n} b^{m+1}\}, m \geq 0, n \geq 0$. Oba tento jazyk prijímajú nasledovne:

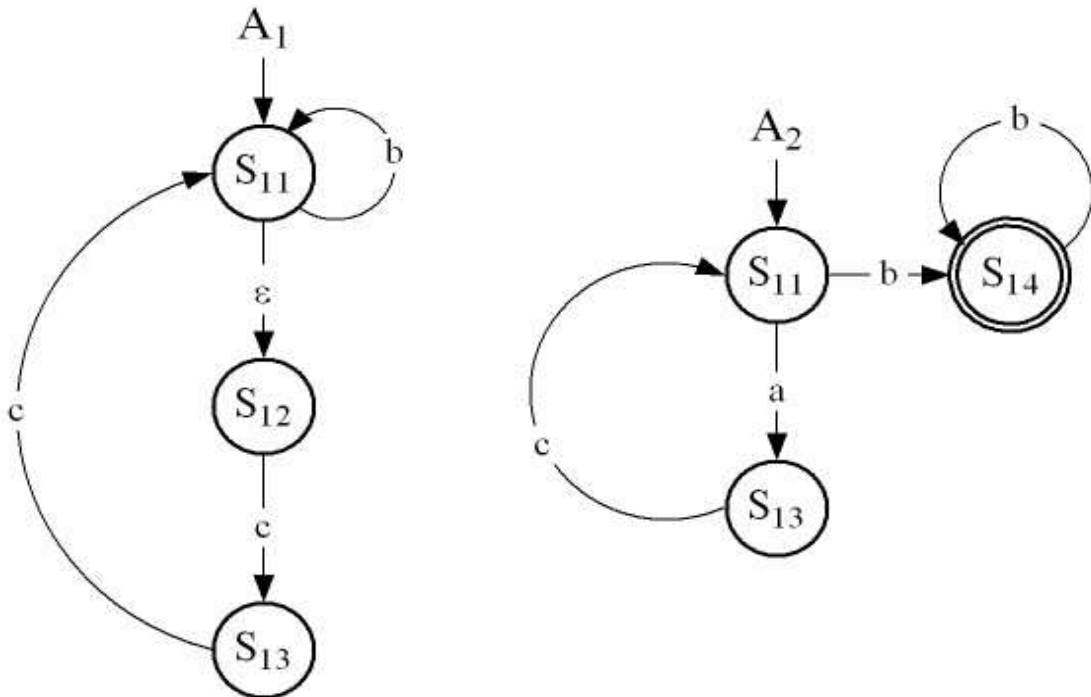
2-SKP:

$$S_{11}bccS_{21}aabb \Rightarrow S_{11}bccS_{21}aabb \Rightarrow S_{12}ccS_{22}abb \Rightarrow S_{12}ccS_{21}bb \Rightarrow S_{13}cS_{23}b \Rightarrow S_{12}S_{23}P.$$

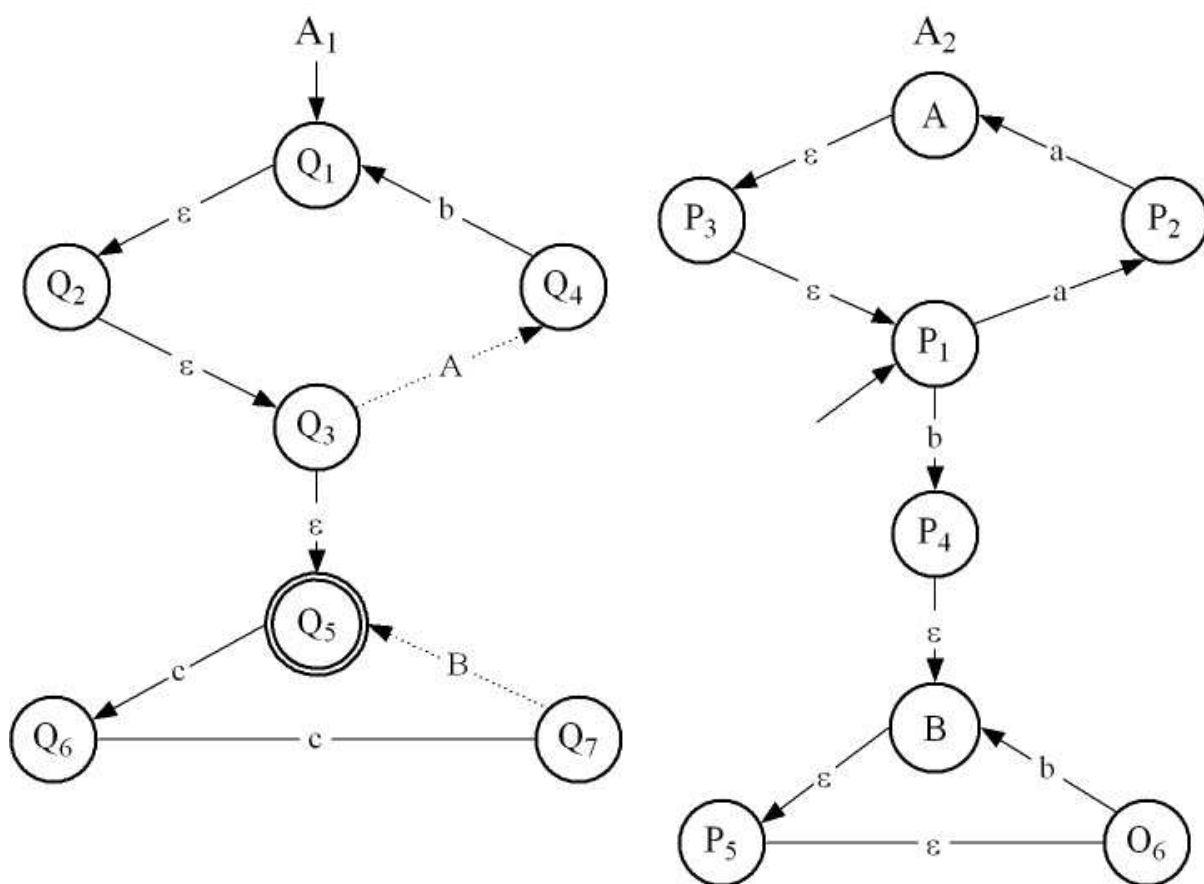
CPC:

$$\begin{aligned} Q_1bccP_1aabb &\Rightarrow Q_2bccP_2abb \Rightarrow Q_3bccAbb \Rightarrow Q_4bccP_3bb \Rightarrow Q_1ccP_1bb \Rightarrow Q_2ccP_4b \Rightarrow \\ &\Rightarrow Q_3ccBb \Rightarrow Q_5ccP_5b \Rightarrow Q_6cP_6b \Rightarrow Q_7B \Rightarrow Q_5P_5. \end{aligned}$$

Na nasledujúcich dvoch obrázkoch je celý priebeh fungovania systému znázornený, na obrázku 2.3 je 2-SKP automatový systém a na obrázku je PC AS FSA automatový systém:



Obrázok 2.3



Obrázok 2.4

Ako na obrázkoch vidieť, tak automatový systém 2-SKP je o poznanie jednoduchší a to najmä kvôli tomu, že CPC AS FSA využíva na riadenie svojej činnosti komunikáciu prechodmi. Inak sú tieto systémy svojou silou v podstate totožné a to aj napriek tomu, že pracujú na rozličných princípoch.

3 Budúci vývoj a uplatnenie v praxi

V tejto práci sú definované dva nové typy automatových systémov, stavovo kontrolovaný paralelný automatový systém a pravidlovo kontrolovaný paralelný automatový systém, ale tieto definície sú iba začiatok, a v budúcnosti môže vývoj týchto a aj ostatných automatových systémov pokračovať.

Tento vývoj môže v nasledujúcich rokoch smerovať rôznymi smermi, pretože oblasť formálnych jazykov a najmä oblasť automatových systémov je iba vo svojich počiatkoch a je stále veľa otvorených možností. Je možné pokračovať napríklad tak, že sa budú skúmať už definované automatové systémy a rôzne ich variácie, ako napríklad automatové systémy s návratom, automatové systémy s odlišným spôsobom komunikácie, poprípade s odlišným spôsobom riadenia činnosti a tieto princípy sa môžu uplatniť pri rozvoji automatových systémov definovaných v tejto práci a teda v budúcnosti môžu vzniknúť rôznorodé automatové systémy, ktoré budú kombinovať vlastnosti viacerých automatových systémov.

Ďalším možným smerom je skúmanie automatových systémov, ktoré budú pozostávať z rôznorodých komponent, ktorými napríklad budú zásobníkové automaty, Turingove stroje, alebo dokonca systémy už zložené z rôznych druhov komponent.

Smer, ktorý má ešte asi najviac možností je skúmanie v oblasti gramatík a gramatických systémov, pretože tejto oblasti bol doposiaľ venovaná väčšia pozornosť ako automatovým systémom a podľa získaných poznatkov navrhovať nové typy automatových systémov.

Aj keď táto práca je zameraná skôr teoreticky tak praktické využitie automatových systémov v nej zavedených je určite široké, napríklad v oblasti prekladačov, kde sa vo veľkom využívajú rôzne typy automatových systémov, ale aj v oblastiach kompilátorov, lingvistiky, ktorá ich môže využívať napríklad pri triedení jazykov, alebo pri hľadaní spoločných znakov viacerých jazykov a podobne, v oblasti mikrobiológie, kde sa môžu automatové systémy uplatniť pri skúmaní správania mikroorganizmov, ich životného cyklu atď. a vzhľadom na to, že automatové systémy v súčasnosti nachádzajú čoraz väčšie uplatnenie, tak nie je v budúcnosti vylúčené ich možné ďalšie využitie.

Záver

V teórii formálnych jazykov sa v poslednom období dostávajú do výraznejšej pozornosti okrem gramatických systémov aj automatové systémy a táto práca sa svojím obsahom rozširuje oblasť poznania práve automatových systémov.

Medzi moderné trendy a to nielen v oblasti formálnych jazykov, ale aj v iných oblastiach patrí paralelný spôsob spracovania úlohy, teda úloha je rozdelená na viacero menších častí, ktoré sa spracovávajú naraz a tento postup je výhodnejší ako spracovávanie celej časti sekvenčne. A týmto smerom sa riadila aj táto práca a automatové systémy v nej definované.

Najskôr boli ukázané niektoré už existujúce automatové systémy, ich princípy na ktorých jednotlivé typy automatových systémov pracujú a boli taktiež ukázané základné princípy kanonických multigeneratívnych systémov a potom na základe týchto poznatkov boli definované dva nové typy automatových systémov a to stavovo kontrolovaný paralelný automatový systém a pravidlovo kontrolovaný paralelný automatový systém. Následne bol ukázaný vzájomný vzťah medzi obidvoma typmi automatových systémov a to algoritmami, ktoré ukazujú ako je možné previesť jeden automatový systém na druhý a zase naopak, následne boli skúmané vlastnosti týchto automatových systémov a taktiež boli v práci definované automatové systémy porovnávané s už existujúcimi automatovými systémami.

Táto práca sa zaoberala automatovými systémami najmä z teoretického hľadiska, teda boli použité definície a algoritmy, ktoré popisovali tieto systémy a tieto boli vhodne doplnené vzorovými príkladmi spolu s ilustráciami, aby bol obsah práce názornejší. Praktické hľadisko bolo načrtnuté v tretej kapitole, s tým, že do budúcnosti sa uplatnenie automatových systémov môže ešte výrazne zväčšiť. Taktiež tretia kapitola ukazuje akým smerom by sa mohol vývoj definovaných automatových systémov, ale aj ostatných automatových systémov uberať a týchto možností je značné množstvo, pretože ešte stále zostáva rozsiahla oblasť, či už automatových, alebo gramatických systémov, ktorá nebola preštudovaná. Tento budúci vývoj sa môže v podstate rozdeliť na dve základné roviny a to na teoretickú, ktorá sa môže zaoberať definovaním a skúmaním ďalších automatových systémov a ich vlastnosťami a na praktickú rovinu, ktorá bude skúmať praktické využitie automatových systémov a študovať, aké ich vlastnosti treba poprípade modifikovať.

Cieľ tejto práce, ktorým bolo preštudovať automatové systémy a na základe získaných poznatkov navrhnúť nový typ automatových systémov a následne skúmať vlastnosti tohto systému a porovnať ho z už existujúcimi systémami bol, podľa môjho názoru splnený a ako osobný prínos pri tvorbe tejto práce vidím detailnejšie zoznámenie s automatovými systémami, respektíve rozšírenie znalostí o oblasti formálnych jazykov a získanie väčšieho prehľadu v tejto oblasti, ktorý môžem v budúcnosti využiť či už pri ďalšom štúdiu, alebo pri výkone povolania.

Ďalšie prípadné rozšírenie tejto práce by sa mohlo vydať cestou naprogramovania určitého či už stavovo, alebo pravidlovo kontrolovaného paralelného automatového systému, s ohľadom na jeho praktické využitie a teda naprogramovať buď to prekladač, alebo kompilátor, poprípade sa pustiť do zložitejšej aplikácie v oblasti lingvistiky, alebo mikrobiológie.

Literatúra

- [1] Češka, M.: Teoretická informatika, učební text FIT VUT v Brně, 2002, [online], [cit.09-05-2008], rev. 07-02-2008], URL<<http://www.fit.vutbr.cz/study/courses/TIN/public/Texty/ti.pdf>>.
- [2] Meduna, A.: Automata and Languages, Theory and Applications, Springer, London, 2000.
- [3] Krithivasan, K., Sakthi Balan, N., Harsha, P.: Distributed processing in automata, in: International Journal of Foundations of Computer Science, Vol. 10, No. 4, 1999, s. 443-463.
- [4] Goehring, M.: Parallel communicating automata systems, učební text University of Potsdam, 2006, [cit. 10-05-2008], [rev. 18-03-2008].
- [5] Martín-Vide, C., Mitrana, V.: Some undecidable problems for parallel communicating finite automata systems, in: Information Processing Letters, Vol. 77, 2001, s. 239-245.
- [6] Martín-Vide, C., Mateescu, A., Mitrana, V.: Parallel finite automata systems communicating by states, in: International Journal of Foundations of Computer Science, Vol. 13, No. 5, 2002, s. 733-749.
- [7] Petřík P.: Automatové systémy, FIT VUT v Brně, Brno, 2007, bakalářská práce.
- [8] Lukáš R.: Multigenerativní gramatické systémy, FIT VUT v Brně, Brno, 2006, disertační práce.
- [9] Dassow, J., Paun, Gh., Rozenberg, G.: Grammar Systems, in: G. Rozenberg and A. Salomaa, (eds.), Handbook of Formal Languages, Vol. 2, Springer, Berlin 1997.

Zoznam príloh

Príloha 1. CD obsahujúce text bakalárskej práce vo formáte pdf.